



Arbitrary High-Order Spline Finite Element Solver for the Time Domain Maxwell equations

Ahmed Ratnani, Eric Sonnendrücker

► To cite this version:

Ahmed Ratnani, Eric Sonnendrücker. Arbitrary High-Order Spline Finite Element Solver for the Time Domain Maxwell equations. Journal of Scientific Computing, 2011, 10.1007/s10915-011-9500-8 . hal-00507758

HAL Id: hal-00507758

<https://hal.science/hal-00507758>

Submitted on 30 Jul 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Arbitrary High-Order Spline Finite Element Solver for the Time Domain Maxwell equations

Ahmed Ratnani*

Eric Sonnendrücker†

July 30, 2010

Abstract

In this paper, we study high order methods for solving the time domain Maxwell equations using spline finite elements on domains defined by NURBS. Convenient basis functions for the discrete exact sequence of spaces introduced by Buffa et al [4] are exhibited which provided the same discrete structure as for classical Whitney Finite Elements. An analysis of stability of the time scheme is also developed.

Keywords. Maxwell's equations, Spline Finite Element method, Isogeometric analysis, Time domain.

AMS subject classification. 65M60

1 Introduction

Since their introduction, B-splines have had a great success, thanks mainly to fast and stable algorithms developed for their use. They are used as well in industry as in academic research for interpolation, data fitting and computer aided design. Recent works by Hughes and co-authors [18, 6, 19] and the introduction of iso-geometric analysis have added yet another dimension to their use, creating an interface between simulation and numerical modeling.

It seems that before the recent works of Hughes [18], the use of splines as basis functions in the finite element method was quite uncommon and essentially limited to uniform B-splines using periodic conditions, even though the web-splines developed by Hoellig and co-workers provided a strategy for dealing with boundary conditions [16, 15]. The idea of iso-geometric analysis using geometric transformations and non uniform splines or NURBS appears much simpler for most applications. Compared to traditional finite elements, the main change due to the iso-geometric analysis is undoubtedly the emergence of the k-refinement, a strategy that can increase the regularity of functions through the mesh's interfaces, to reduce the number of degrees of freedom.

Modern finite element techniques for Maxwell's equations rely on ideas from differential geometry and more precisely on the existence of discrete spaces that provide an exact De Rham sequence. Following pioneering ideas by Bossavit [2, 3] a complete theory was successively developed [14, 1]. Buffa and co-workers [4, 5] extended iso-geometric analysis to steady-state Maxwell's equations providing a discrete exact De Rham sequence involving discrete spaces based on B-splines.

*INRIA Nancy Grand Est, project-team CALVI and IRMA, UMR 7501 Université de Strasbourg - CNRS

†IRMA, UMR 7501 Université de Strasbourg - CNRS, and project-team CALVI, INRIA Nancy Grand Est, 7 rue René Descartes 67084 STRASBOURG Cedex (`{ratnani, sonnen}@math.unistra.fr`)

Our aim is to use this discrete sequence for the solution of the time-dependent Maxwell equations. One important feature of geometric Finite Maxwell solvers is that one of Faraday's or Ampere's law hold strongly in the discrete spaces and the other one needs a Galerkin Hodge operator [3]. Hence, one of the discrete equations is completely explicit and the other one involves a mass matrix that yields a linear system to be solved at each iteration. Thanks to a property of B-spline derivatives, we were able to exhibit basis functions of the discrete spaces involved in the De Rham sequence, such that the same property holds and the discrete curl or divergence, depending on the chosen formulation, is simply an incidence matrix depending only on the connectivity of the mesh, which is thus independent of the geometric transformation. We have implemented this idea in 2D, but as it is based on a tensor product approximation, it extends straightforwardly to 3D.

The outline of the paper is the following. First we recall the variational formulation of the 2D Maxwell equation, then we give the most important properties of B-splines and iso-geometric analysis. After that, we construct our exact sequence of discrete spaces as spans of well chosen basis functions on a cartesian grid and explain how to transform them on the geometric domain defined by NURBS. We can then compute the semi-discrete equations in space in a matrix formulation. We then prove a stability condition when a Leap-Frog algorithm in time is used. Finally, different test cases are performed to validate the method. High-order convergence is obtained and CFL conditions are computed for different degrees of splines.

2 Variational formulation for the 2D Maxwell equations

We consider the Maxwell equations in a subdomain $\Omega \subset \mathbb{R}^2$ with a regular boundary denoted by $\Gamma = \partial\Omega$. We note \mathbf{n} , the outward unit normal vector of Ω on the boundary Γ . We recall that in 2D, we have two curl operators, one acting on scalars $\mathbf{curl} u = \begin{pmatrix} \frac{\partial u}{\partial y} \\ -\frac{\partial u}{\partial x} \end{pmatrix}$, and one acting on vectors $\mathbf{v} = \begin{pmatrix} v_x \\ v_y \end{pmatrix}$ for which $\mathbf{curl} \mathbf{v} = \frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y}$. The divergence of a vector \mathbf{v} is defined by $\mathbf{div} \mathbf{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}$.

We shall also need the following function spaces

$$H(\mathbf{curl}, \Omega) = \{ \mathbf{v} \in (L^2(\Omega))^2; \mathbf{curl} \mathbf{v} \in L^2(\Omega) \} \text{ and } H_0(\mathbf{curl}, \Omega) = \{ \mathbf{v} \in H(\mathbf{curl}, \Omega); \mathbf{v} \times \mathbf{n} = 0 \},$$

$$H(\mathbf{div}, \Omega) = \{ \mathbf{v} \in (L^2(\Omega))^2; \mathbf{div} \mathbf{v} \in L^2(\Omega) \}.$$

Notice that the space that could be called

$$H(\mathbf{curl}, \Omega) = \{ u \in (L^2(\Omega))^2; \mathbf{curl} u \in L^2(\Omega) \}$$

is identical to $H^1(\Omega)$. We shall therefore stick with the more usual $H^1(\Omega)$.

Finally, we recall the Green formula we will need:

$$\int_{\Omega} (\mathbf{curl} G) \cdot \mathbf{F} dX = \int_{\Omega} G \mathbf{curl} \mathbf{F} dX - \int_{\Gamma} (G \times \mathbf{n}) \cdot \mathbf{F} dS, \forall \mathbf{F} \in H(\mathbf{curl}, \Omega), \forall G \in H^1(\Omega), \quad (1)$$

and

$$\int_{\Omega} (\mathbf{div} \mathbf{F}) G dX = - \int_{\Omega} \mathbf{F} \cdot (\nabla G) dX + \int_{\Gamma} \mathbf{F} \cdot \mathbf{n} G dS, \forall \mathbf{F} \in H(\mathbf{div}, \Omega), \forall G \in H^1(\Omega). \quad (2)$$

In 2D domains, Maxwell's equations can be decoupled into two systems. The first involving the (E_x, E_y, H_z) components is called the Transverse Electric (TE) mode, and the second, involving

the (H_x, H_y, E_z) components is called the Transverse Magnetic (TM) mode. As both mode can be discretized in the same manner, we shall only consider in the sequel the TE mode which reads

$$\frac{\partial \mathbf{E}}{\partial t} - \text{curl } H = -\mathbf{J}, \quad (3)$$

$$\frac{\partial H}{\partial t} + \text{curl } \mathbf{E} = 0, \quad (4)$$

$$\text{div } \mathbf{E} = \rho, \quad (5)$$

where the components are defined by $\mathbf{E} = \begin{pmatrix} E_x \\ E_y \end{pmatrix}$, $H = H_z$. These equations need to be supplemented with initial and boundary conditions. We shall only consider periodic or perfectly conducting boundary conditions $\mathbf{E} \times \mathbf{n} = 0$ and in a second step Silver-Müller absorbing boundary conditions.

In order to derive a conforming Finite Element approximation of Maxwell's equations we first need to write an appropriate variational formulation. We would like to stay with the first order version of the system and are then naturally led to a mixed formulation involving two different functional spaces for \mathbf{E} and H . The two options are, after multiplying both equations by a test function and integrating by parts, to use Green's formula for either one of the two equations but not for both.

The first variational formulation, in the case of perfectly conduction boundary conditions, can be derived using Green's formula (1) in Ampere's law (3). This yields

Find $(\mathbf{E}, H) \in H_0(\text{curl}, \Omega) \times L^2(\Omega)$ such that

$$\frac{d}{dt} \int_{\Omega} \mathbf{E} \cdot \boldsymbol{\psi} \, dX - \int_{\Omega} H(\text{curl } \boldsymbol{\psi}) \, dX = - \int_{\Omega} \mathbf{J} \cdot \boldsymbol{\psi} \, dX \quad \forall \boldsymbol{\psi} \in H_0(\text{curl}, \Omega), \quad (6)$$

$$\frac{d}{dt} \int_{\Omega} H \varphi \, dX + \int_{\Omega} (\text{curl } \mathbf{E}) \varphi \, dX = 0 \quad \forall \varphi \in L^2(\Omega). \quad (7)$$

Using the Green formula (1) in Faraday's law (4) yields the second variational formulation

Find $(\mathbf{E}, H) \in H(\text{div}, \Omega) \times H^1(\Omega)$ such that

$$\frac{d}{dt} \int_{\Omega} \mathbf{E} \cdot \boldsymbol{\psi} \, dX - \int_{\Omega} (\text{curl } H) \cdot \boldsymbol{\psi} \, dX = - \int_{\Omega} \mathbf{J} \cdot \boldsymbol{\psi} \, dX \quad \forall \boldsymbol{\psi} \in H(\text{div}, \Omega), \quad (8)$$

$$\frac{d}{dt} \int_{\Omega} H \varphi \, dX + \int_{\Omega} \mathbf{E} \cdot (\text{curl } \varphi) \, dX = 0 \quad \forall \varphi \in H^1(\Omega). \quad (9)$$

Notice that in the first variational (6)-(7) formulation the boundary condition is taken into account in strong form by putting it into the function space where \mathbf{E} lives. On the other hand, in the second formulation (8)-(9) the boundary condition is taken into account in a weak form.

3 Splines and B-splines functions

3.1 Splines

Splines are piecewise polynomials defined on the real line. We shall require that on each compact interval, they consist of a small number of non vanishing polynomial pieces.

Let $T^* = \{t_i^*, 0 \leq i \leq s\}$ be a finite strictly increasing sequence of points of \mathbb{R} . A function S on \mathbb{R} is a spline of order $k, k \geq 1$ with the breakpoints T^* if on each interval (t_i^*, t_{i+1}^*) , it is a polynomial of degree $\leq p := k - 1$. On the other hand, the spline can have any regularity less than $p - 1$ at the breakpoints. The smoothness r_i of a spline S at the breakpoint t_i^* is defined as follows:

- $r_i := 0$ if S is discontinuous at t_i^* , otherwise
- r_i is the largest integer $0 < r_i \leq k$ so that S has continuous derivatives of orders $< r_i$

therefore, we denote the associated spline space on an interval $I = [a, b]$, $a := t_0^*$, $b = t_s^*$ by $S_k^*(T^*, I)$ which consists of all splines S of order $\leq k$ with breakpoints contained in T^* and of smoothness $\geq r_i$ at t_i^* .

Rather than use the smoothness of the spline at breakpoints, we use the defect $m_i := k - r_i$, this is the number of degrees of freedom of S at t_i^* . A simple computation leads to $\dim S_k(T^*, \mathbf{m}, I) = k + \sum_{i=0}^s m_i$ with $\mathbf{r} := (m_0, \dots, m_s)$. The space $S_k(T^*, \mathbf{m}, I)$ is called the Schoenberg space. A simple basis of the Schoenberg space is

$$S_{-j}(x) = \frac{(x-a)^j}{j!}, \quad j = 0, \dots, k$$

$$S_{i,j}(x) = \frac{(x-t_i^*)_+^j}{j!}, \quad j = k - m_i, \dots, k-1, \quad i = 1, \dots, s$$

Curry and Schoenberg [7] have introduced another basis which has a more local character. Their basis are splines with the smallest possible support. They are defined by means of the divided differences and called basic splines (B-splines). For more details on this subject we refer to the books of De-Boor [8] (for computational aspect), DeVore and Lorentz [9] (for more theoretical aspects).

3.2 B-Splines

Let $T = (t_i)_{1 \leq i \leq N+k}$ be a non-decreasing sequence of knots.

Definition 1 (B-Spline) The i -th B-Spline of order k is defined by the recurrence relation:

$$N_j^k = w_j^k N_j^{k-1} + (1 - w_{j+1}^k) N_{j+1}^{k-1}$$

where,

$$w_j^k(x) = \frac{x - t_j}{t_{j+k-1} - t_j} \quad N_j^1(x) = \chi_{[t_j, t_{j+1}[}(x)$$

We note some important properties of a B-splines basis:

- B-splines are piecewise polynomial of degree $p = k - 1$
- Positivity
- Compact support; the support of N_j^k is contained in $[t_j, \dots, t_{j+k}]$
- Partition of unity: $\sum_{i=1}^N N_i^k(x) = 1, \forall x \in \mathbb{R}$
- Local linear independence
- If a knot t has a multiplicity m then the B-spline is $\mathcal{C}^{(p-m)}$ at t

For each knot vector $T = (t_i)_{1 \leq i \leq N+k}$ we will associate the breakpoints sequence $(t_i^*)_{1 \leq i \leq s}$. Let $(P_i)_{1 \leq i \leq N} \in \mathbb{R}^d$ be a sequence of control points, forming a control polygon.

Definition 2 (B-Spline curve) The B-spline curve in \mathbb{R}^d associated to $T = (t_i)_{1 \leq i \leq N+k}$ and $(P_i)_{1 \leq i \leq N}$ is defined by:

$$\mathbf{M}(t) = \sum_{i=1}^N N_i^k(t) \mathbf{P}_i$$

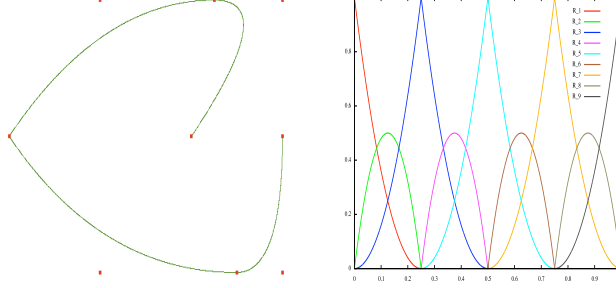


Figure 1: (left) A B-spline curve and its control points, (right) B-splines functions used to draw the curve. $N = 9$, $p = 2$, $T = \{000, \frac{1}{4}\frac{1}{4}, \frac{1}{2}\frac{1}{2}, \frac{3}{4}\frac{3}{4}, 111\}$

3.3 Fundamental geometric operations

After modification, we denote by $\tilde{N}, \tilde{k}, \tilde{T}$ the new parameters. (\mathbf{Q}_i) are the new control points.

3.3.1 Knot insertion

One can insert a new knot t , where $t_j \leq t < t_{j+1}$. For this purpose we use the DeBoor algorithm:

$$\begin{aligned} \tilde{N} &= N + 1 \\ \tilde{k} &= k \\ \tilde{T} &= \{t_1, \dots, t_j, t, t_{j+1}, \dots, t_{N+k}\} \\ \alpha_i &= \begin{cases} 1 & 1 \leq i \leq j - k + 1 \\ \frac{t - t_i}{t_{i+k-1} - t_i} & j - k + 2 \leq i \leq j \\ 0 & j + 1 \leq i \end{cases} \\ \mathbf{Q}_i &= \alpha_i \mathbf{P}_i + (1 - \alpha_i) \mathbf{P}_{i-1} \end{aligned}$$

3.3.2 Order elevation

We can elevate the order of the basis, without changing the curve. Several algorithms exist for this purpose. We used the one by Huang et al. [17].

$$\begin{aligned} \tilde{k} &= k + m \\ \tilde{m}_i &= m_i + m \\ \tilde{N} &= N + ms \end{aligned}$$

Differential coefficients are defined as $\tilde{\mathbf{P}}_i^l$:

$$\tilde{\mathbf{P}}_i^l = \begin{cases} \tilde{\mathbf{P}}_i & l = 0 \\ \frac{1}{t_{i+p}-t_{i+l}}(\tilde{\mathbf{P}}_{i+1}^{l-1} - \tilde{\mathbf{P}}_i^{l-1}) & l > 0, t_{i+k-1} > t_{i+l} \\ 0 & l > 0, t_{i+k-1} = t_{i+l} \end{cases}$$

$\beta_i = \sum_{l=1}^i m_l, 1 \leq i \leq s-1$, et $\alpha_i = \prod_{l=1}^i \frac{k-1-l}{k-1+m-l}, 1 \leq i \leq k-2$

We present the algorithm by [17]:

1. Compute $\tilde{\mathbf{P}}_0^j, 0 \leq j \leq k-1$ et $\tilde{\mathbf{P}}_{\beta_l}^i, 1 \leq l \leq s-1, k-m_l \leq i \leq k-1$
2. Compute $\tilde{\mathbf{Q}}_0^j = \prod_{l=1}^j (\frac{k-l}{k+m-l}) \tilde{\mathbf{P}}_0^j, 0 \leq j \leq k-1$
3. Compute $\tilde{\mathbf{Q}}_{\beta_l+m_l}^j = \prod_{l=1}^j (\frac{k-l}{k+m-l}) \tilde{\mathbf{P}}_{\beta_l}^j, 1 \leq l \leq s-1, k-m_l \leq i \leq k-1$
4. Compute $\tilde{\mathbf{Q}}_{\beta_l+m_l+i}^{k-1} = \tilde{\mathbf{Q}}_{\beta_l+m_l}^{(k-1)}, 1 \leq l \leq s-1, 1 \leq i \leq m$
5. Compute $\tilde{\mathbf{Q}}_i^0$

Note that there exist other algorithms such those given by (see [22, 21] and others). The one given in [17] is more efficient and much more simple to implement. We can also use a more sophisticated version of this algorithm to do the insertion of new knots while elevating the degree.

3.4 Refinement strategies

Refining the grid can be done in 3 different ways. This is the most interesting aspects of B-splines basis.

- using the patch parameter h , by inserting new knots. This is the h-refinement, it is the equivalent of mesh refinement of the classical finite element method.
- using the degree p , by elevating the B-spline degree. This is the p-refinement, it is the equivalent of using higher finite element order in the classical FEM.
- using the regularity of B-splines, by increasing / decreasing the multiplicity of inserted knots. This is the k -refinement. This new strategy does not have an equivalent in the classical FEM.

J.A. Evans et al. [11] studied the k-refinement using the theory of Kolmogorov n-widths. As we will see in this article, the use of this strategy can be more efficient than the classical p-refinement, as it reduces the dimension of the basis.

3.5 Grid generation

For this purpose, we use alternatively h and p -refinement. The minimal degree of the basis functions is imposed by the domain design. When inserting knots, we can use uniformly-spaced knots or non uniformly-spaced ones.

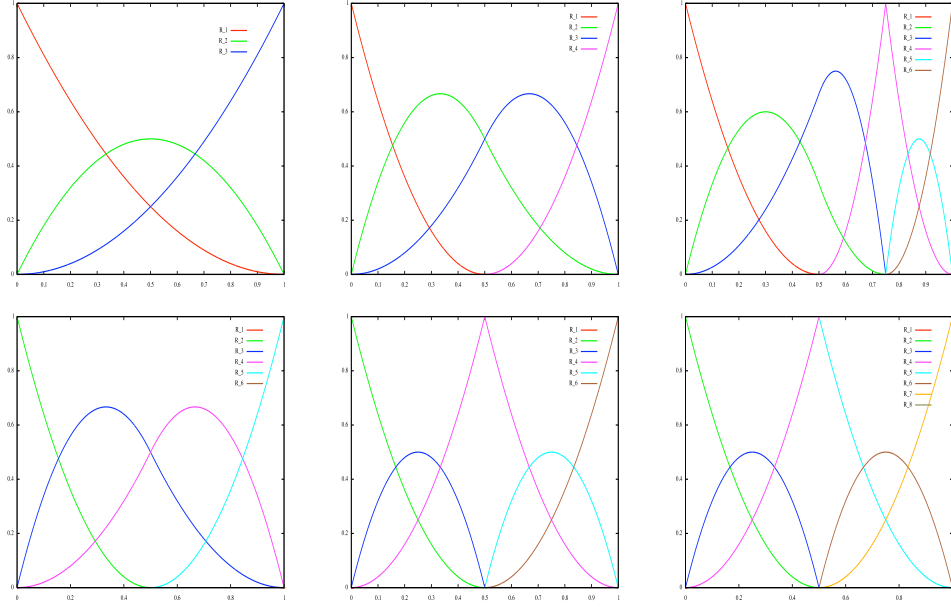


Figure 2: First line, using h -refinement with $p = 2$, $T = \{000, 111\}$, $T = \{000, \frac{1}{2}, 111\}$ and $T = \{000, \frac{1}{2}, \frac{3}{4}, 111\}$. Second line, using p and h -refinement with $p = 3$, $T = \{0000, \frac{1}{2}, 1111\}$, $T = \{0000, \frac{1}{2}, \frac{1}{2}, 1111\}$ and $T = \{0000, \frac{1}{2}, \frac{1}{2}, 1111\}$

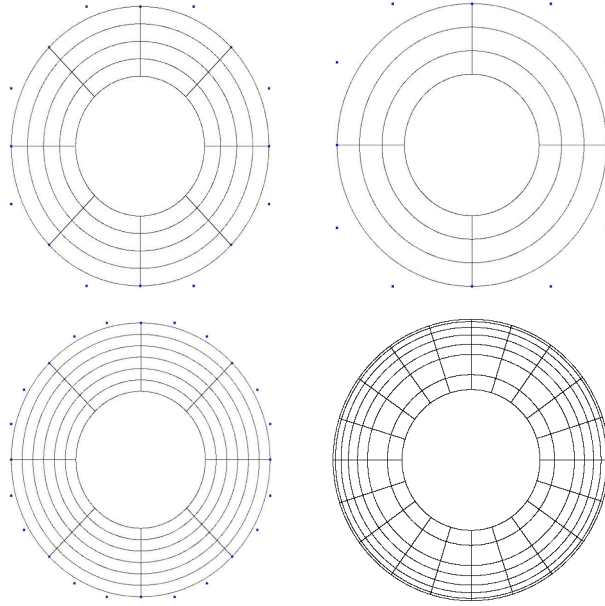


Figure 3: Grid generation. 1st line: (left) after h -refinement, $N_1 = 17$, $N_2 = 5$, (right) after p -refinement, $p_1 = p_2 = 3$. 2nd line: (left) after h -refinement $p_1 = p_2 = 3$, (right) using unstructured mesh, $p_1 = p_2 = 2$.

3.6 Spline finite elements on general domains

In order to use the spline finite elements on general domains, we shall use the ideas of isogeometric analysis and consider domains defined with nurbs that can be obtained via CAD software and use these nurbs to map a rectangular domain on the physical domain.

Let Q be a cell in the physical domain. \tilde{Q} is the parametric associated cell and such that $Q = F(\tilde{Q})$. let J_F be the Jacobian of the transformation F , that maps any parametric domain point (ξ, η) into physical domain point (x, y) .

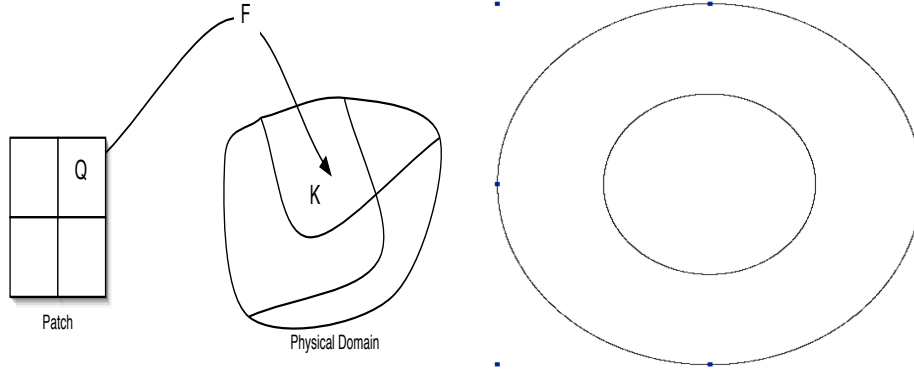


Figure 4: Mapping from the patch to the physical domain: (left) initial patch, (right) patch after h-refinement in the η direction

For any function v of (x, y) we associate its representation in the parametric domain

$$\tilde{v}((\xi, \eta)) := v \circ F((\xi, \eta)) = v((x, y)).$$

The basis functions R_i will not be affected by these changes, the reader can always know if the we are working in the physical or parametric domain thanks to

$$(x, y) = F(\xi, \eta), x = \alpha(\xi, \eta) \text{ and } y = \beta(\xi, \eta).$$

then,

$$\alpha_1 = \frac{\partial \alpha}{\partial \xi} \quad \alpha_2 = \frac{\partial \alpha}{\partial \eta} \quad \beta_1 = \frac{\partial \beta}{\partial \xi} \quad \beta_2 = \frac{\partial \beta}{\partial \eta},$$

we have for the determinant of the Jacobian $\det(J_F) = \alpha_1\beta_2 - \alpha_2\beta_1$ and for $J_{F^{-1}}$

$$J_F = \begin{pmatrix} \alpha_1 & \alpha_2 \\ \beta_1 & \beta_2 \end{pmatrix} \quad J_{F^{-1}} = \frac{1}{\Delta} \begin{pmatrix} \beta_2 & -\alpha_2 \\ -\beta_1 & \alpha_1 \end{pmatrix}.$$

Let u be a (scalar or vector) function defined on the physical domain. When we use the patch coordinates, we will write it \tilde{u} , *idem* for the used spaces.

3.7 Boundary conditions

The boundary conditions are very easy to handle. Thanks to the mapping, the boundary of the physical domain is mapped into the boundary of the patch. therefore, it becomes to treat periodic condition, by using periodic splines with an adapted knot vector [8]. The perfect conductor and Silver-Muller conditions are also simple to implement.

4 Construction of the finite element spaces

An important feature of the functional spaces we chose for the variational formulation is that they form an exact sequence. Depending of the variational formulation we choose, we need to work with different exact sequences. In the case of (6)-(7), the following function spaces are involved

$$\begin{array}{ccccc} & \mathbf{grad} & & \mathbf{curl} & \\ H^1(\Omega) & \longrightarrow & H(\mathbf{curl}, \Omega) & \longrightarrow & L^2(\Omega) \\ \cup & & \cup & & \cup \\ V & \longrightarrow & W_{curl} & \longrightarrow & X \end{array} \quad (10)$$

In the case of (8)-(9), the following function spaces are involved

$$\begin{array}{ccccc} & \mathbf{curl} & & \mathbf{div} & \\ H^1(\Omega) & \longrightarrow & H(\mathbf{div}, \Omega) & \longrightarrow & L^2(\Omega) \\ \cup & & \cup & & \cup \\ V & \longrightarrow & W_{div} & \longrightarrow & X \end{array} \quad (11)$$

In order to keep the specific features of Maxwell's equations at the discrete level we need to construct finite dimensional subspaces endowed with the same structure. The involved discrete spaces are denoted by $X \subset H^1(\Omega)$, $W_{curl} \subset H(\mathbf{curl}, \Omega)$, $W_{div} \subset H(\mathbf{div}, \Omega)$ and $V \subset L^2(\Omega)$. When discretizing the first variational formulation (6)-(7), we shall look for $(\mathbf{E}_h, H_h) \in W_{curl} \times V$ and when discretizing the second variational formulation (8)-(9), we shall look for $(\mathbf{E}_h, H_h) \in W_{div} \times X$.

4.1 Spline finite elements on patch grids

We shall now start constructing the actual subspaces $X \subset H^1(\Omega)$, $W_{curl} \subset H(\mathbf{curl}, \Omega)$, $W_{div} \subset H(\mathbf{div}, \Omega)$ and $V \subset L^2(\Omega)$.

Our discrete space will be constructed using B-splines.

The key point of our method is the use of the recursion formula for the derivatives:

$$N_i^{p'}(t) = p \left(\frac{N_i^{p-1}(t)}{t_{i+p} - t_i} - \frac{N_{i+1}^{p-1}(t)}{t_{i+p+1} - t_{i+1}} \right). \quad (12)$$

It will be convenient to introduce the notation $D_i^p = p \frac{N_i^{p-1}(t)}{t_{i+p} - t_i}$. Then the recursion formula for derivative simply becomes

$$N_i^{p'}(t) = D_i^p(t) - D_{i+1}^p(t). \quad (13)$$

Discrete vector fields on a rectangular domain. Let us first consider a rectangular domain Ω . We consider the following discrete functional spaces

$$\begin{aligned} V &= \text{span}\{N_i^p(x)N_j^p(y), 1 \leq i \leq N_x, 1 \leq j \leq N_y\}, \\ W_{div} &= \text{span}\left\{\begin{pmatrix} N_i^p(x)D_j^p(y) \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ D_i^p(x)N_j^p(y) \end{pmatrix}, 1 \leq i \leq N_x, 1 \leq j \leq N_y\right\}, \\ W_{curl} &= \text{span}\left\{\begin{pmatrix} D_i^p(x)N_j^p(y) \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ N_i^p(x)D_j^p(y) \end{pmatrix}, 1 \leq i \leq N_x, 1 \leq j \leq N_y\right\}, \\ X &= \text{span}\{D_i^p(x)D_j^p(y), 1 \leq i \leq N_x, 1 \leq j \leq N_y\}. \end{aligned}$$

As proved in the article by Buffa et al, [4] these spaces verify the same exact property as the spaces they approximate.

Vector field transformations. Let us now define coordinate changes conserving either the curl or the divergence of a vector field.

Let us start from a vector field $\underline{\Psi}(\xi, \eta) = (\underline{\Psi}^{(1)}(\xi, \eta), \underline{\Psi}^{(2)}(\xi, \eta))^T$ defined on the parametric domain \tilde{Q} .

Using the transformation formula (18) for the vector fields of W_{div} which conserves the divergence, using the diffeomorphism $G = F^{-1}$, we get $\Psi^{(1)} = \frac{1}{\Delta}(\alpha_2 \Psi^{(2)} + \alpha_1 \Psi^{(1)})$ and $\Psi^{(2)} = \frac{1}{\Delta}(\beta_2 \Psi^{(2)} + \beta_1 \Psi^{(1)})$.

So let $\Psi = (\Psi^{(1)}, \Psi^{(2)})^T$ be a function in W_{div} , and $\underline{\Psi} = (\underline{\Psi}^{(1)}, \underline{\Psi}^{(2)})^T$ be a function in \widetilde{W}_{div} . To save the divergence property, the corresponding space of \widetilde{W}_{div} on the physical domain is

$$W_{div} = \{ \Psi := (\frac{1}{\Delta}(\alpha_1 \Psi^{(1)} + \alpha_2 \Psi^{(2)}), \frac{1}{\Delta}(\beta_1 \Psi^{(1)} + \beta_2 \Psi^{(2)}))^T, \quad \underline{\Psi} \in \widetilde{W}_{div} \}$$

therefore,

$$W_{div} = \text{span} \left\{ \frac{1}{\Delta} \tilde{N}_i^p(\xi) \tilde{D}_j^p(\eta) \begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix}, \frac{1}{\Delta} \tilde{D}_i^p(\xi) \tilde{N}_j^p(\eta) \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix} \right\}.$$

4.2 The Discrete Equations

Let us now express the equation satisfied by the approximations \mathbf{E}_h, H_h when using each of the variational formulations we introduced. Let's start with (8)-(9). In this case we look for $\mathbf{E}_h \in W_{div}$ and $H_h \in H^1(\Omega)$. We first notice that due to the exact sequence property we have $\text{div} \mathbf{E}_h \in X$.

Let us denote by

$$\psi_{i,j}^1 = \begin{pmatrix} N_i^p(x) D_j^{p-1}(y) \\ 0 \end{pmatrix}, \quad \psi_{i,j}^2 = \begin{pmatrix} 0 \\ D_i^{p-1}(x) N_j^p(y) \end{pmatrix}$$

we have

$$W_{div} = \text{span} \{ \psi_{i,j}^1, \psi_{i,j}^2, \quad 1 \leq i \leq N_x, 1 \leq j \leq N_y \},$$

Denoting the components of \mathbf{E}_h by (E_h^x, E_h^y) , we have

$$E_h^x(t, x, y) = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} e_{i,j}^x(t) N_i^p(x) D_j^p(y), \quad E_h^y(t, x, y) = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} e_{i,j}^y(t) D_i^p(x) N_j^p(y),$$

and

$$H_h^z(t, x, y) = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} h_{i,j}^z(t) N_i^p(x) N_j^p(y).$$

Using these expansions on the finite element bases, we can compute explicitly, for

$$\mathbf{curl} H_h = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} h_{i,j}^z(t) \begin{pmatrix} N_i^p(x) N_j^{p'}(y) \\ -N_i^{p'}(x) N_j^p(y) \end{pmatrix}$$

we now use the formula (12),

$$\begin{aligned} \mathbf{curl} H_h &= \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} h_{i,j}^z(t) \begin{pmatrix} N_i^p(x) (D_j^{p-1}(y) - D_{j+1}^{p-1}(y)) \\ -(D_i^{p-1}(x) - D_{i+1}^{p-1}(x)) N_j^p(y) \end{pmatrix} \\ &= \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} h_{i,j}^z(t) \{ \psi_{i,j}^1 - \psi_{i,j+1}^1 - \psi_{i,j}^2 + \psi_{i+1,j}^2 \} \end{aligned}$$

We will use the following convention: for a knot vector $T = (t_i)_{1 \leq i \leq N+k}$ generating the "N" B-splines of order k , $\{N_i^k, 1 \leq i \leq N\}$, we put $N_j^k := 0$ for any $j > N$. By making a change of index in the sum, we get

$$\mathbf{curl} H_h = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} h_{i,j}^z \{\psi_{i,j}^1 - \psi_{i,j}^2\} - \sum_{i=1}^{N_x} \sum_{j=2}^{N_y+1} h_{i,j-1}^z \psi_{i,j}^1 + \sum_{i=2}^{N_x+1} \sum_{j=1}^{N_y} h_{i-1,j}^z \psi_{i,j}^2$$

therefore,

$$\begin{aligned} \mathbf{curl} H_h &= \sum_{i=2}^{N_x} \sum_{j=2}^{N_y} (h_{i,j}^z - h_{i,j-1}^z) \psi_{i,j}^1 - (h_{i,j}^z - h_{i-1,j}^z) \psi_{i,j}^2 \\ &+ \sum_{j=2}^{N_y} \{h_{1,j}^z - h_{1,j-1}^z\} \psi_{1,j}^1 + \sum_{i=2}^{N_x} \{h_{i-1,1}^z - h_{i,1}^z\} \psi_{i,1}^2 + h_{1,1}^z \psi_{1,1}^1 - h_{1,1}^z \psi_{1,1}^2 \end{aligned}$$

in case of periodic boundary conditions, we get a strong form of the discrete Ampere's law (8) which can be written

$$-\dot{\mathbf{e}} + R h^z = M_W^{-1} j, \quad (14)$$

where h^z , e^x , e^y the vectors of spline coefficients, M_W is the mass matrix for W_{div} and R consists of two blocks of the discrete derivatives in the x and y directions of these vectors.

Remark We've shown that we can write $K^T = M_W R$, where K is the matrix involved in the classical formulation (without courant field) :

$$\begin{cases} M_W \dot{\mathbf{e}} = K h \\ M_V \dot{h} = -K^T \mathbf{e} \end{cases}$$

A discrete Faraday's law can be obtained from the variational formulation (7) which can be written

$$\dot{h}^z + d_x e^y - d_y e^x = 0, \quad (15)$$

where d_x and d_y the discrete derivatives in the x and y directions.

We have thus obtained matrix differential equations on the vectors of spline coefficients that can be solved by any appropriate ODE solver. For example, for a second order time discretization, a leap-frog or Verlet scheme is well adapted.

4.3 Solving strategy

The use of Isogeometric analysis for solving the Maxwell's equations is not immediate. The problem is the De Rham diagram that requires certain regularity of the domain which is logic. Let us illustrate our purpose.

Let $T_p^{(1)}, T_p^{(2)}, 1 \leq i, j \leq N + p + 1$ be two open vectors knots, where we choose $N_1 = N_2 = N$ and $p_1 = p_2 = p$ for simplicity. To get the discrete De Rham diagram, the discrete spaces involved in our scheme must be:

$$\mathcal{S}_{\alpha,\alpha}^{p,p}(\mathcal{P}) \subset H^1(\Omega), \quad \mathcal{S}_{\alpha,\alpha-1}^{p,p-1} \times \mathcal{S}_{\alpha-1,\alpha}^{p-1,p}(\mathcal{P}) \subset H(div, \Omega)$$

where $\alpha = \{-1, m, \dots, p - m, -1\}$ is the smoothness of the Spline space at the breakpoints, *i.e* each interior knot has a multiplicity m .

Let us assume that the domain, using appropriate control points, is generated by the knots vectors $T_{p-1}^{(1)}, T_{p-1}^{(2)}$. If we denote by m the multiplicity of interior knots, we see that after the elevation degree process, we will have the vectors $T_p^{(1)}, T_p^{(2)}$ with a multiplicity $m + 1$ of the interiors knots. therefore, some parts (at the interior knots) of our domain must be of a regularity $p - m$. This can be done by suppressing the redundant knot. Remark, that when using NURBS this is a more difficult process.

We now present the strategy for solving Maxwell's equation using B-splines:

- we model exactly our domain, using NURBS, B-splines or any other mapping, which can maps the physical domain into a rectangular domain "patch",
- if the domain can not be exactly represented, we can interpolate, or approximate the physical domain using B-splines or NURBS, [20, 13],
- the discrete spaces are generated by B-splines such they can respect the exact discrete De Rham sequence.

5 Leap Frog scheme's stability

In this section we will give a preview of the time discretization scheme we used to solve the linear system obtained and some classical results. Let us first, just recall the general formulation of the LF scheme. For the linear system

$$\begin{cases} M_W \dot{\mathbf{e}} = K \mathbf{h} \\ M_V \dot{\mathbf{h}} = -K^T \mathbf{e} \end{cases}$$

the LF time discretization of order N is given by

$$\begin{cases} M_W \frac{E^{n+1} - E^n}{\Delta t} = K_N H^{n+\frac{1}{2}} \\ M_V \frac{H^{n+\frac{3}{2}} - H^{n+\frac{1}{2}}}{\Delta t} = -K_N^T E^{n+1} \end{cases}$$

where

$$\begin{cases} K_N = K & , N = 2 \\ K_N = K(I - \frac{\Delta t^2}{24} M_W^{-1} K M_V^{-1} K^T) & , N = 4 \end{cases}$$

We define the global energy by $\mathcal{E}^n := \frac{1}{2} \{ (E^n)^T M_W E^n + (H^{n-\frac{1}{2}})^T M_V H^{n+\frac{1}{2}} \}$ We have the following lemma:

Lemma 1 The global energy is stationary, *i.e* $\mathcal{E}^{n+1} = \mathcal{E}^n$

Proof Let us put $E^{n+\frac{1}{2}} = \frac{E^{n+1} + E^n}{2}$. Then we have,

$$\begin{aligned} \mathcal{E}^{n+1} - \mathcal{E}^n &= \frac{1}{2} ((E^{n+1})^T M_W E^{n+1} + (H^{n+\frac{1}{2}})^T M_V H^{n+\frac{3}{2}}) - \frac{1}{2} ((E^n)^T M_W E^n + (H^{n-\frac{1}{2}})^T M_V H^{n+\frac{1}{2}}) \\ &= (E^{n+1})^T M_W E^{n+\frac{1}{2}} - (E^n)^T M_W E^{n+\frac{1}{2}} + \frac{1}{2} (H^{n-\frac{1}{2}})^T M_V (H^{n+\frac{3}{2}} - H^{n+\frac{1}{2}}) \\ &= (E^{n+\frac{1}{2}})^T M_W (E^{n+1} - E^n) + \frac{1}{2} (H^{n-\frac{1}{2}})^T M_V (H^{n+\frac{3}{2}} - H^{n+\frac{1}{2}}) \end{aligned}$$

using the time discretization scheme, we have:

$$\mathcal{E}^{n+1} - \mathcal{E}^n = \Delta t (E^{n+\frac{1}{2}})^T K_N H^{n+\frac{1}{2}} - \Delta t (H^{n+\frac{1}{2}})^T K_N^T E^{n+\frac{1}{2}}$$

Therefore, we have $\mathcal{E}^{n+1} - \mathcal{E}^n = 0$

The stability of the scheme depends on the global energy, it must be a positive quadratic form

Lemma 2 \mathcal{E}^n is a positive quadratic form if $\Delta t \leq \frac{2}{d_N}$, where $d_N = \|M_V^{-\frac{1}{2}} K_N^T M_W^{-\frac{1}{2}}\|$

Proof

$$\begin{aligned}
\mathcal{E}^n &= \frac{1}{2}(E^n)^T M_W E^n + \frac{1}{2}(H^{n-\frac{1}{2}})^T M_V H^{n+\frac{1}{2}} = \frac{1}{2}(E^n)^T M_W E^n + (H^{n-\frac{1}{2}})^T M_V H^{n-\frac{1}{2}} - \frac{\Delta t}{2}(H^{n-\frac{1}{2}})^T M_V E^n \\
&\geq \frac{1}{2}\|M_W^{\frac{1}{2}}\|^2 + \frac{1}{2}\|M_V^{\frac{1}{2}}\|^2 - \frac{\Delta}{2}|(H^{n-\frac{1}{2}})^T M_V^{\frac{1}{2}} M_V^{-\frac{1}{2}T} K_N M_W^{-\frac{1}{2}} M_W^{\frac{1}{2}} E^n| \\
&\geq \frac{1}{2}\|M_W^{\frac{1}{2}}\|^2 + \frac{1}{2}\|M_V^{\frac{1}{2}}\|^2 - \frac{\Delta d_N}{2}\{\|M_V^{\frac{1}{2}} H^{n-\frac{1}{2}}\|^2 + \|M_W^{\frac{1}{2}} E^n\|^2\} \\
&\geq \frac{1}{2}\|M_W^{\frac{1}{2}}\|^2 + \frac{1}{2}\|M_V^{\frac{1}{2}}\|^2 - \frac{\Delta d_N}{4}\|M_V^{\frac{1}{2}} H^{n-\frac{1}{2}}\| \|M_W^{\frac{1}{2}} E^n\| \\
&\geq \frac{1}{2}(1 - \frac{\Delta t d_N}{2})\{\|M_W^{\frac{1}{2}}\|^2 + \|M_V^{\frac{1}{2}}\|^2\}
\end{aligned}$$

The last quantity is positive if $1 \geq \frac{\Delta t d_N}{2}$.

Finally, we can take $CFL_N^{th} = \frac{2}{hd_N}$. As proved in [12], we have $CFL_4^{th} \simeq 2.85 CFL_2^{th}$.

6 Numerical results

The theoretical convergence rate is " $p+1$ " for the magnetic field H^z and " p " for the electrical field \mathbf{E} , where $p = \min(p_1, p_2)$.

6.1 Test case 1: square

The analytical solution in this case is:

$$\begin{aligned}
H^z &= \cos(k_1 x + \phi_1) \cos(k_2 y + \phi_2) \cos(\omega t) \\
E^x &= -\frac{k_2}{\omega} \cos(k_1 x + \phi_1) \sin(k_2 y + \phi_2) \sin(\omega t) \\
E^y &= \frac{k_2}{\omega} \sin(k_1 x + \phi_1) \cos(k_2 y + \phi_2) \sin(\omega t)
\end{aligned}$$

For our test we took a computational domain of size $[0, 2\pi] \times [0, 2\pi]$ and

$$k_1 = k_2 = 1, \quad \phi_1 = \phi_2 = 0, \quad \omega = \pi$$

This test enables to check the L^2 norm of the error (in log scale) between the numerical and analytical solution, with respect to the parameter $h = \max\{diam(\tilde{Q})\}$, for different orders of the basis functions (from order 3 to order 6). This is shown in figure 5. We verify that the slopes of the different curves correspond to the order of the basis functions. In particular, for high orders, the machine precision is achieved. For validation, we solved the Maxwell's equation for $N = 16, 32, 64, 128$, with $dt = \frac{1}{100}, \frac{1}{200}, \frac{1}{400}, \frac{1}{800}$. The L^2 error is then computed after respectively 5, 10, 20, 40 iterations so as to keep the same final time.

Thanks to the regularity of the basis functions, we see that we can achieve a good precision with a lower number of meshes. In other words, the k-refinement strategy helped as to reduce

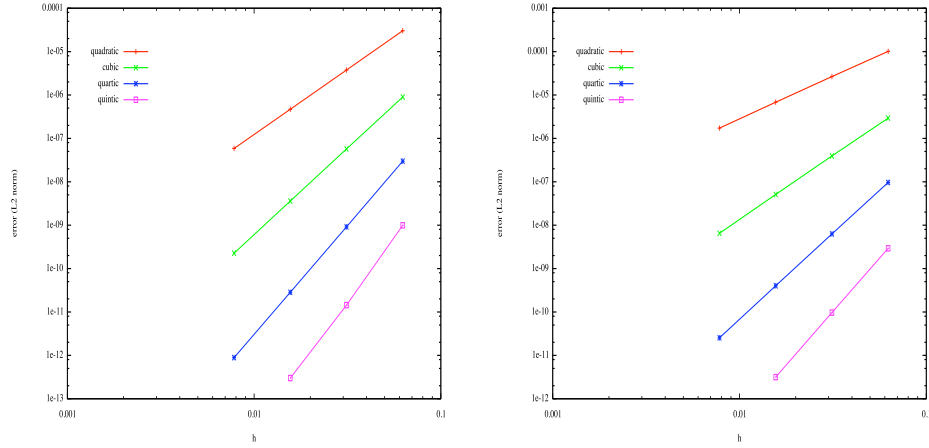


Figure 5: Square test: the L^2 norm error for (left) the magnetic field , (right) the electric field

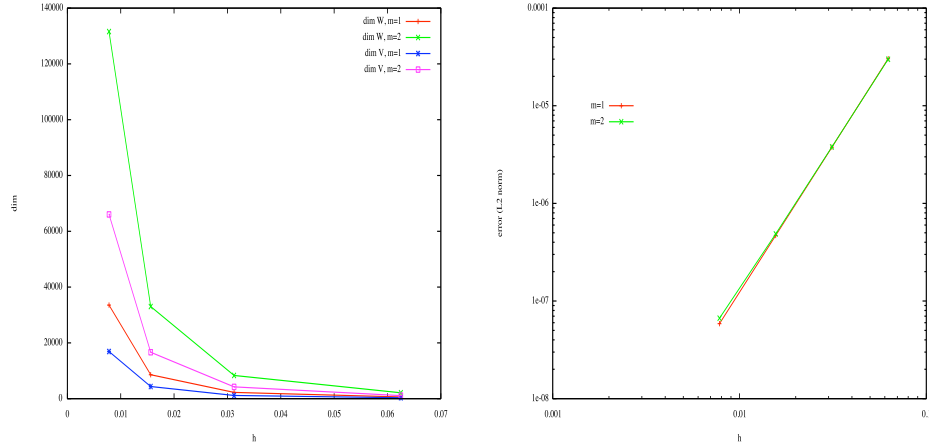


Figure 6: Square test: (left) the dimension of the discrete spaces W_h and V_h , (right) the L^2 norm error for the electric field, where the vector knots are multiplicity $m = 1, 2$ for quadratic B-splines

the number of degrees of freedom as we can see it in the figure 6, where the number of degrees of freedom was reduced by a factor of 6. The price to pay, is that we increased the support of the basis functions compared to the classical finite element method. But as we said before, this is the best we can do using splines functions, the B-splines have the minimal support that we can get if we try to increase the regularity of our basis.

Tables (1,2) show the CFL numbers for different B-splines degree $p = 2, \dots, 5$. We see that the CFL decreases with the B-splines degree and the knots multiplicity.

6.2 Test case 2: circular wave guide

The analytical solution, in the polar coordinates, in this case is:

	$LF2^{Th}$	$LF2^{num}$	$LF4^{Th}$	$LF4^{num}$
$p = 2$	0.3044	0.3056	0.8676	0.8720
$p = 3$	0.2058	0.1840	0.5866	0.5872
$p = 4$	0.1496	0.1520	0.4265	0.4272
$p = 5$	0.1151	0.1168	0.3281	0.3280

Table 1: Test case 1: CFL numbers (theoretic and numerical values), for splines of degree $p = 2, \dots, 5$

	m=1	m=2
$p = 2$	0.8720	0.5200
$p = 3$	0.5872	0.4224
$p = 4$	0.4272	0.3056
$p = 5$	0.3280	0.2304

Table 2: Test case 1: CFL, using LF4, for splines of degree $p = 2, \dots, 5$ for singular knots ($m = 1$), and doubled knots ($m = 2$)

$$\begin{aligned}
H^z(r, \theta) &= -\cos(\omega t + \theta) \{J_1(\omega r) + aY_1(\omega r)\} \\
E^\theta(r, \theta) &= \frac{1}{2} \sin(\omega t + \theta) \{J_0(\omega r) - J_2(\omega r) + a(Y_0(\omega r) - Y_2(\omega r))\} \\
E^r(r, \theta) &= -\frac{1}{\omega r} \cos(\omega t + \theta) \{J_1(\omega r) + aY_1(\omega r)\}
\end{aligned}$$

where J_n, Y_n are the first and the second Bessel functions of order n .
For our test we took

$$r_{min} = 0.65138750344695903414, \quad r_{max} = 0.99000418530735846839, \quad a = 1.0, \quad \omega = 3\pi$$

We used uniform periodic B-splines in the theta direction and uniform B-splines with open knots in the radial direction. The use of uniform B-splines allows as to reduce the number of degrees of freedom. therefore, we can achieve an error of 10^{-11} , using quintic B-splines, with only 64×64 meshes, and $\dim W_h = 8640, \dim V_h = 4352$. A precision of 10^{-7} is achieved, using quintic B-splines, with only 16×16 meshes.

6.3 Test case 3: Silver-Muller condition

In the case of the Silver-Muller condition the discrete Faraday's equation of (9) is written

$$\partial_t M_V h^z + K \mathbf{e} + \Gamma h^z = 0$$

where Γ is the mass matrix of the discrete V_h on the boundary that checks Silver-Muller condition *i.e* $\int_{\partial\Omega_{SM}} N_{i,j} N_{i',j'}$

In this test we see the evolution of an electromagnetic wave (figure 6.3), under Silver-Muller condition on both the internal and external boundary. At $t = 0$ we took $E_x = u(x)u(y)'$ and $E_Y = u(x)'u(y)$, with $u(x) = \exp(-\frac{(x-m)^2}{2\sigma^2})$.

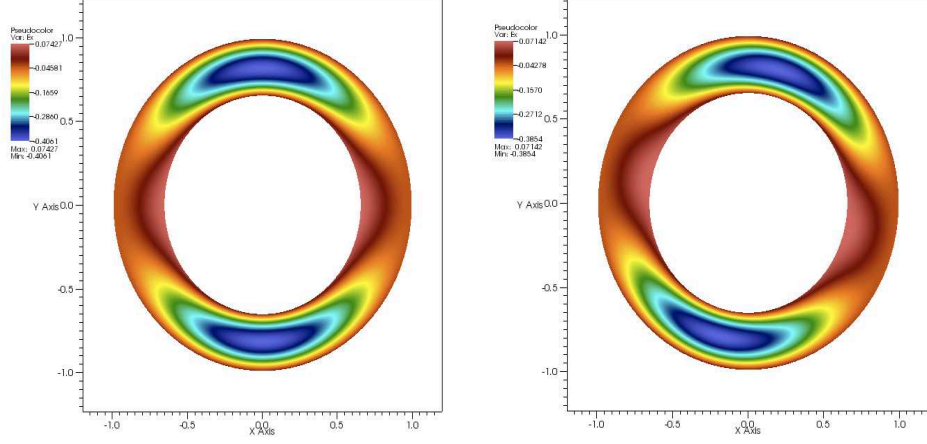


Figure 7: Circular wave guide test: (left) at $t = 0$, (right) after 20 iterations, for $N = 64, p = 3$

7 Conclusions and perspectives

We presented a new scheme for Maxwell's equation using B-splines functions, which enables as to reduce the number of degrees of freedom thanks to the k-refinement. Our method can be easy used with an integrated CAD system. However, the use of the Isogeometric idea requires to verify the De Rham diagram, so, we can not always use an exact modeling. therefore, we will need to approximate or interpolate domains using either splines or NURBS functions. Another idea, which can be a great help is the use of GB-Splines. GB-Splines are a general basic splines, and verify a closed relation to (12):

$$G'_{i,k}(t) = \frac{G_{i,k-1}(t)}{c_{i,k-1}} - \frac{G_{i+1,k-1}(t)}{c_{i+1,k-1}}, \quad k \geq 3$$

with

$$G_{i,2}(t) = \begin{cases} \psi_{i,n}^{(n-2)}(t), & t_i^* \leq t \leq t_{i+1}^* \\ \phi_{i+1,n}^{(n-2)}(t), & t_{i+1}^* \leq t \leq t_{i+2}^* \\ 0, & t \notin (t_i^*, t_{i+2}^*) \end{cases}$$

and $c_{i,k-1} = \int_{t_i^*}^{t_{i+k-1}^*} G_{i,k-1}(t)dt$, $\psi_{i,n}^{(n-2)}, \phi_{i,n}^{(n-2)}$ are assumed to be strictly monotone on (t_i^*, t_{i+1}^*) and (t_{i+1}^*, t_{i+2}^*) respectively. We can get the classical polynomial spline of order n by taking

$$\phi_{i,n}(t) = -\frac{(t - t_i^*)^{n-1}}{(n-1)!h_i}, \quad \psi_{i,n}(t) = \frac{(t - t_i^*)^{n-1}}{(n-1)!h_i}, \quad h_i = t_{i+1}^* - t_i^*$$

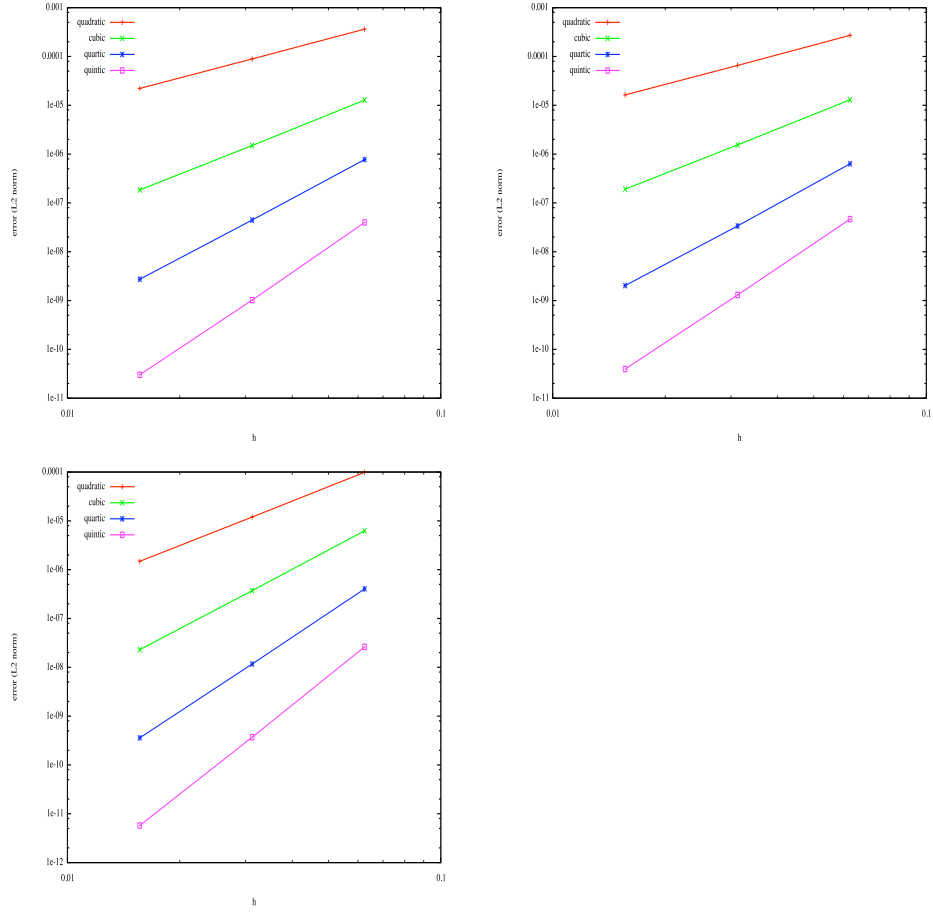


Figure 8: Circular wave guide test: the L^2 norm error for, first line E^x (left) and E^y (right) components of the electric field, second line the magnetic field

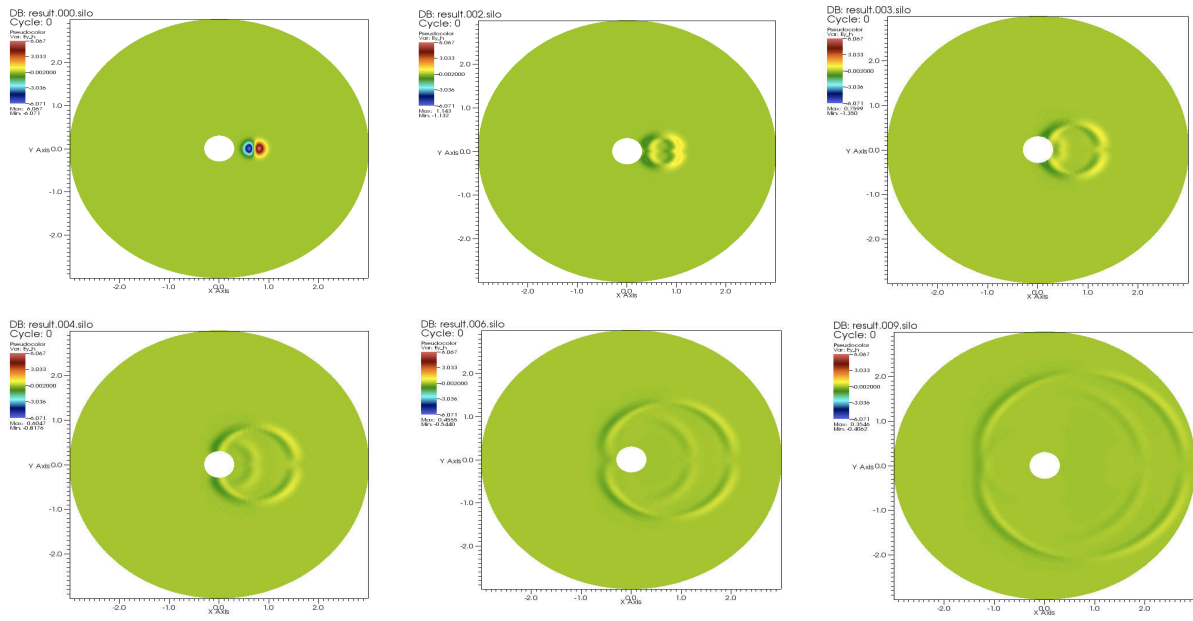


Figure 9: Evolution of an electromagnetic wave in circular domain under Silver-Muller boundary condition

A Transformation compatible with grad, div and curl operators

In order to define our basis functions on the parametric domain, which is a rectangular domain of \mathbb{R}^2 with cartesian coordinates and then to map them onto a patch of the physical domain, we need to define a transformation of scalar and vector fields which is compatible with our differential operators (grad, div and curl). This is provided to us by the pullback operator for differential forms which is designed to commute with the exterior derivative. Hence compatible transformations will be provided to us by associating our scalar or vector fields to a well chosen differential form and using the pullback.

In our case, we have differential forms defined on Q and need to construct the associated differential forms on K . For this we need a C^1 diffeomorphism $G : K \rightarrow Q$. Let us recall the pullback formula for 0, 1 and 2-forms. A 0-form on Q is a function (or a scalar field) $\varphi(\xi, \eta)$. The pullback of φ on K is in this case simply $\varphi = \varphi \circ G$.

A 1-form on the parametric space can be written $\underline{\omega} = \underline{\omega}_1(\xi, \eta)d\xi + \underline{\omega}_2(\xi, \eta)d\eta$. Denoting by G_1 and G_2 the components of the diffeomorphism G , the pullback of $\underline{\omega}$ on K is then defined by

$$\begin{aligned} \omega &= G_* \underline{\omega} = \underline{\omega}_1 \circ G dG_1 + \underline{\omega}_2 \circ G dG_2 \\ &= \left(\omega_1 \frac{\partial G_1}{\partial x} + \omega_2 \frac{\partial G_2}{\partial x} \right) dx + \left(\omega_1 \frac{\partial G_1}{\partial y} + \omega_2 \frac{\partial G_2}{\partial y} \right) dy, \end{aligned} \quad (16)$$

where we denote by $\omega_1(x, y) = \underline{\omega}_1 \circ G(x, y)$ and $\omega_2(x, y) = \underline{\omega}_2 \circ G(x, y)$.

A 2-form on the parametric space can be written $\underline{\sigma}(\xi, \eta) d\xi \wedge d\eta$ and its pullback on K by the diffeomorphism G is defined by

$$\begin{aligned} \sigma(x, y) dx \wedge dy &= G_*(\underline{\sigma} d\xi \wedge d\eta) = \underline{\sigma} \circ G dG_1 \wedge dG_2 \\ &= \sigma(x, y) \left(\frac{\partial G_1}{\partial x} \frac{\partial G_2}{\partial y} - \frac{\partial G_2}{\partial x} \frac{\partial G_1}{\partial y} \right) dx \wedge dy, \end{aligned} \quad (17)$$

where we denote by $\sigma(x, y) = \underline{\sigma} \circ G(x, y)$.

Now a vector field in a 2D space is associated to a differential 1-form. This 1-form depends on the sequence of spaces we are working on and is chosen such that its exterior derivative corresponds either to the curl or the divergence of the vector field. Note that in both cases a function φ (or scalar field) can be associated to either a 0-form which is the function itself or the two form $\varphi d\xi \wedge d\eta$.

Let us start with the case of (10). In this case the exterior derivative of a 0-form should be associated to the **grad** operator and the exterior derivative of a 1-form should be associated to the curl operator. This is the case if we associate a generic vector field $\underline{\Psi}(\xi, \eta) = (\underline{\Psi}^{(1)}(\xi, \eta), \underline{\Psi}^{(2)}(\xi, \eta))^T$ to the differential form

$$\underline{\omega}_c = \underline{\Psi}^{(1)}(\xi, \eta) d\xi + \underline{\Psi}^{(2)}(\xi, \eta) d\eta.$$

Indeed, take a function (or 0-form) φ . Then, on the one hand

$$\mathbf{grad} \varphi = (\partial_\xi \varphi, \partial_\eta \varphi)^T$$

which is associated to the one form

$$\partial_\xi \varphi d\xi + \partial_\eta \varphi d\eta = d\varphi,$$

an on the other hand

$$\text{curl } \underline{\Psi}(\xi, \eta) d\xi \wedge d\eta = (\partial_\xi \underline{\Psi}^{(2)} - \partial_\eta \underline{\Psi}^{(1)}) d\xi \wedge d\eta = d\underline{\omega}_c.$$

Let us now consider the case of (11). Then the exterior derivative of a 0-form should be associated to the **curl** operator and the exterior derivative of a 1-form should be associated to the div operator. This is the case if we associate a generic vector field $\underline{\Psi}(\xi, \eta) = (\underline{\Psi}^{(1)}(\xi, \eta), \underline{\Psi}^{(2)}(\xi, \eta))^T$ to the differential form

$$\underline{\omega}_d = \underline{\Psi}^{(1)}(\xi, \eta) d\eta - \underline{\Psi}^{(2)}(\xi, \eta) d\xi.$$

Indeed, take a function (or 0-form) $\underline{\varphi}$. Then, on the one hand

$$\mathbf{curl} \underline{\varphi} = (\partial_\eta \underline{\varphi}, -\partial_\xi \underline{\varphi})^T$$

which is associated to the one form

$$\partial_\eta \underline{\varphi} d\eta - (-\partial_\xi \underline{\varphi} d\xi) = d\underline{\varphi},$$

an on the other hand

$$\text{div } \underline{\Psi}(\xi, \eta) d\xi \wedge d\eta = (\partial_\xi \underline{\Psi}^{(1)} + \partial_\eta \underline{\Psi}^{(2)}) d\xi \wedge d\eta = d\underline{\omega}_d.$$

Having now associated our functions and vector fields to differential forms, we can use the expression of the pullbacks to define the adequate scalar and vector field transformations.

In particular, when using the spaces associated to (11). We need to transform the basis functions associated to V , this is straightforward as they are functions associated to 0-forms, and to W_{div} . To defined the transformation of a vector field $\underline{\Psi}(\xi, \eta) = (\underline{\Psi}^{(1)}(\xi, \eta), \underline{\Psi}^{(2)}(\xi, \eta))^T \in W_{div}$ we use the pullback formula (16) for the 1-form $\underline{\omega}_d = \underline{\Psi}^{(1)}(\xi, \eta) d\eta - \underline{\Psi}^{(2)}(\xi, \eta) d\xi$. This yields

$$\omega_d = \left(-\underline{\Psi}^{(2)} \circ G \frac{\partial G_1}{\partial x} + \underline{\Psi}^{(1)} \circ G \frac{\partial G_2}{\partial x} \right) dx + \left(-\underline{\Psi}^{(2)} \circ G \frac{\partial G_1}{\partial y} + \underline{\Psi}^{(1)} \circ G \frac{\partial G_2}{\partial y} \right) dy,$$

which is associated to the vector field

$$\Psi = \left(\left(-\underline{\Psi}^{(2)} \circ G \frac{\partial G_1}{\partial y} + \underline{\Psi}^{(1)} \circ G \frac{\partial G_2}{\partial y} \right), \left(\underline{\Psi}^{(2)} \circ G \frac{\partial G_1}{\partial x} - \underline{\Psi}^{(1)} \circ G \frac{\partial G_2}{\partial x} \right) \right)^T. \quad (18)$$

References

- [1] D. N. ARNOLD, R. S. FALK, R. WINTHER, *Finite element exterior calculus, homological techniques, and applications*. Acta Numer. 15 (2006), pp 1–155.
- [2] A. BOSSAVIT, *Mixed finite elements and the complex of Whitney forms*, The Mathematics of Finite Elements and Applications VI, J. Whiteman (ed.), 137–144, Academic Press, London, 1988.
- [3] A. BOSSAVIT, *Computational electromagnetism and geometry. (5): The 'Galerkin hodge'*, J. Japan Soc. Appl. Electromagn. & Mech., 8, 2 (2000), pp. 203–209.
- [4] A. BUFFA, G. SANGALLI, R. VAZQUEZ, *Isogeometric analysis in electromagnetics: B-splines approximation*, Comput. Methods Appl. Mech. Engrg. **199**, pp. 1143–1152, (2009).
- [5] A. BUFFA, J. RIVAS, G. SANGALLI, R. VÁZQUEZ, *Isogeometric Analysis in Electromagnetics: theory and testing*. Preprint.
- [6] J.A COTTRELL, T. HUGHES, Y. BAZILEVS, *Isogeometric Analysis, toward Integration of CAD and FEA*, first ed., John Wiley & Sons, Ltd, (2009).
- [7] H.B. CURRY, I.J. SCHOENBERG *On Polya frequency functions IV: The fundamental spline functions and their limits*, J. d'Analyse Math. 17, 71-107; Ch. 5:137, 141 (1966).

- [8] C. DE BOOR, *A practical guide to splines*, Springer-Verlag, Berlin, Heidelberg, vol 27 (2001).
- [9] R.A. DEVORE, G.G. LORENTZ, *Constructive Approximation*, Springer-Verlag, Berlin, Heidelberg, (1993).
- [10] M.R. DÖRFEL, B. JÜTTLER, B. SIMEON, *Adaptive isogeometric analysis by local h-refinement with T-splines*, Comput. Methods Appl. Mech. Engrg. **199**, pp. 264–275, (2010).
- [11] J.A. EVANS, Y. BAZILEVS, I. BABUSKA, T. HUGHES, *n-Widths, supinfs, and optimality ratios for the k-version of the isogeometric finite element method*, Comput. Methods Appl. Mech. Engrg. **198**, pp. 1726–1741, (2009).
- [12] H. FAHS, S. LANTERI, *A high-order non-conforming discontinuous Galerkin method for time-domain electromagnetics*, J. Comput. Appl. Math., Vol. 234, pp. 1088-1096 (2010).
- [13] W. HEIDRICH, R. BARTELS, G. LABAHN, *Fitting Uncertain Data with NURBS*, in Proc. 3rd Int. Conf. on Curves and Surfaces in Geometric Design, Vanderbilt University Press, (1996).
- [14] R. HIPTMAIR, *Finite elements in computational electromagnetism*, Acta Numerica 11, pp. 237–339, (2002).
- [15] K. HÖLLIG, *Finite Element Methods with B-Splines*. Frontiers in Applied Mathematics 26, SIAM, 2003.
- [16] K.HÖLLIG, U.REIF, J.WIPPER, *Weighted extended B-spline approximation of Dirichlet problems*. SIAM J. Numer. Anal. 39-2, pp. 442–462 (2001).
- [17] Q.X. HUANG, S.M. HUA, R.R. MARTIN, *Fast degree elevation and knot insertion for B-spline curves*, Comput. Aided Geom. Design **22**, pp. 183–197, (2005).
- [18] T. HUGHES, J. A. COTTRELL, Y. BAZILEVS, *Analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*, Comput. Methods Appl. Mech. Engrg. **194**, pp. 4135-4195, (2005).
- [19] T. HUGHES, A. REALI, G. SANGALLI, *Efficient quadrature for NURBS-based isogeometric analysis*, Comput. Methods Appl. Mech. Engrg. **199**, pp. 301–313, (2010).
- [20] W. MA, J.P. KRUTH, *NURBS Curve and Surface Fitting for Reverse Engineering*, Int. J. Adv. Manuf. Technol. **14**, pp. 918–927, (1998).
- [21] L. PIEGL, W. TILLER, *The NURBS Book*, second ed., Springer-Verlag, Berlin, Heidelberg, (1995).
- [22] H. PRAUTZSCH, B. PIPER, *A fast algorithm to raise the degree of B-spline curves*, Computer Aided Geometric Design **4**, pp. 253–266, (1991).
- [23] T.W. SEDERBERG, D.L. CARDON, J. ZHENG, T. LYCHE, *T-spline Simplification and Local Refinement*, ACM Trans, Graphics 23 (3) pp. 276–283, (2004).